SYSTEM AND METHOD FOR THE AUTOMATED PRESENTATION OF SYSTEM DATA TO, AND INTERACTION WITH, A COMPUTER MAINTAINED DATABASE

CROSS REFERENCE TO RELATED APPLICATIONS

This application claims the priority and benefit of U.S. Provisional Application No. 60/176,535, as well as U.S. Provisional Application No. ____________________, entitled "SYSTEM AND METHOD FOR THE AUTOMATED PRESENTATION OF HEALTH DATA TO, AND ITS INTERACTION WITH, A COMPUTER MAINTAINED DATABASE, TO GENERATE INFORMATION REGARDING POSSIBLE REMEDIES, THERAPIES, PROBLEM SOLUTIONS AND BENEFICIAL PRACTICES, TO IMPROVE USER HEALTH" filed on September 15, 2000, Attorney Docket No. 2761.100, Sidney M. Baker, Inventor, the disclosures of each of which are hereby incorporated herein in their entirety.

BACKGROUND OF THE INVENTION

1. FIELD OF THE INVENTION

The present invention relates generally to the fields of data mining, expert systems, and system theory. In particular, the preferred embodiment relates to interactive data mining regarding the health of a human organism, described as a system.

General System Theory was introduced in the early twentieth century by the German/Canadian Biologist Ludwig Von Bertalanffy. Classical science, and its diverse disciplines, be they chemistry, biology, psychology, or the social sciences, tended to isolate individual *elements* of the observed universe, such as chemical compounds and enzymes, cells, elementary sensations, freely competing individuals, etc. and assumed that by putting theses elements together again, either conceptually or experimentally the whole or system under consideration - - i.e., the cell, mind, or society - - would result and be intelligible. In engineering terminology this approach was equivalent to reducing every system to the linear response of its various components and superposing or aggregating those linear responses to monitor the system as a whole. The problem with such an approach, or opistimology, is the fact that a whole is often more than the sum of its parts. There is often nonlinear and non-intuitive interaction and interdependence between the so called "components" of any system. General system theory is the scientific exploration of wholes and wholeness. General system theory assumes that for a true understanding of any system comprehension not only of the elements is required but of their varied interaction and interrelations as well. This requires exploration of systems in their own right and specificities.

The application of general systems theory to medicine would require nonlinear medical thinking. It mostly has to do with the approach one takes towards understanding what has

caused and event, such as a symptom or a collection of symptoms, signs, and lab tests which are referred to as an illness. As present most medical thinking remains linear. Doctors and patents alike are tempted by the idea that an illness has a single cause that can be treated with a single remedy; such as a pill or a surgical procedure. General systems theory, when applied to medicine, presents ideas about causality in which a web of interactions produces a result that is not easy to pin on a single causative facture. Therefore the resolution of medical problems, or health is sustained by achieving a state of balance among countless strands of the web of genetic, physiologic, psychic, developmental, environmental factors all of which contribute to the state of well being, or lack thereof of human beings. When something goes wrong with ones health, it makes sense to pay attention of all aspects of this web that can be addressed with reasonable cost and risk.

The notion of systems is not unknown to traditional medical thinking. However, its meaning is quite different from the sense it is acquired among the inheritance of general systems theory. Traditionally, medical education is organized via various bodily systems such as the cardiovascular, nervous, immune, reproductive, gastrointestinal, integumentary (skin), musculoskeletal, endocrine, reticuloendothelial and hematologic. It is theses systems that serve as the basis for classifying disease. Upon graduation from medical school novice doctors are expected to choose a particular system and become a specialist. On the other hand, systems theory as applied to medicine provides a unifying model of how things operate, and allows the viewing of biological systems as interconnected and interacting unity of their various components. As a result, one can make functional - - as opposed to

anatomical – divisions, as overall balances assessed within the system. The theory that has dominated medical science for the greater part of the twentieth century is that people get sick because they are the victims of disease. A better theory is that people get sick because of a disruption of the dynamic balance that exists between themselves and their environment. This latter theory works just as well to describe what happens when one gets chicken pox as it does when there is a more complex problem in which many genetic, environmental, and nutritional factors interact.

Because of the prevailing disease oriented approach of medical language the illusion is created that if one possesses the name of a disease responsible for a patients complaints, then one can solve that patients health problem. A better mental model would be one in which all of the details of a person's problem are preserved as opposed to abstracting our theoretical based notions of important as opposed to unimportant "symptoms". Such a language would allow the totality of the information content of the state of a person's health at a given time be preserved. All that would remain needed is the means to extract it and to analyze it.

Digital computers are particularly adapted to such a task. Portraits of a human health status, including reported symptoms, observant indications and laboratory reports can be constructed in such a way so as to preserve the totality of information contained in such a health "snapshot" while still using the names commonly used in medical science to describe the main features of illness. Computers are utilized to make complex pictures out of human health data. If the data is detailed, accurate and structured, the pictures will reflect reality and allow patterns to emerge which are not necessarily visible to the naked eye. The

computer can be used as a "microscope" for viewing large patterns as much as the microscope is used to view the exceedingly small.

In order to use a digital computer in such a way, a format must be created that can be easily encoded into digital data, processed, and decoded into a meaningful output. Users' verbal descriptions of their medical states must be carefully guided into precise and orthogonal categories which can each be assigned a number value, resulting in a multidimensional set of numbers representative of each user's health snapshot. Each dimension would represent some medical attribute. The presence of absence of some condition, sensation, or state, the severity, frequency, or character of the condition, and the duration, onset in correlation to other states or user activities of the problem, to name some general examples.

RELATED ART

Related art in the field of the invention is sparse. Although there are numerous medical database/medical information computer programs and websites, accessible via a local computer, the Internet or other data network, all offering the user the ability to search for a variety of information, none offers the user an opportunity to express the totality of his or her current health snapshot using system provided categories and divisions of the semantic plane. As a result these sites function as efficient and highly accessible medical encyclopedias. Noting more. There is no actual interaction between knowledge stored in the websites server and the health snapshot of the user to generate information that the user would not otherwise know.

In fact, across the gambit of medical web sites and related and equilivent interactive informational tools, the "mental map" or "semantic plane" and the corresponding technical language or taxonomy, by means of which both the queries are posed to, and the information, or output is generated from, the system database - - is the traditional disease based singular cause and effect model discussed above. Therefore, one can at these sites and their equilivent, learn the "causes" and treatments, of a variety of "diseases". As well, one can learn the "disease" causing ones reported symptomology usually, but one cannot discover what percentage of other persons reporting similar symptomology also have similar problems as the user which are not commonly considered to be part of the symptomology of the "disease". For example, suppose someone reports a shortness of breath. Because the medical informational tools currently available to the public do not dynamically interact with the information reported by a user (to the extent that they extensively query the user at all) a given user cannot know that eighty three percent (83%) of persons reporting or seeking the assistance of the medical website also had a strange rash on the soles of their feet. Or, as another example, persons reporting shortness of breath could acquire a variety of information about cardiovascular health and potential problems, but could never know how many people reported a folic acid deficiency and poor night vision as well.

It is only through the articulation of the totality of events (in reality a reasonable tractable representative set thereof) indicative of a human organisms health, including the various mental, biochemical, physical and other processes that completely describes the system as a whole that ones health "system" can be objectively described.

What is therefore desired or needed to truly exploit the massive automated

information extraction and handling and processing capabilities of the digital computer, and by extension, a network of digital computers, is the creation of (i). A carefully constructed taxonomy that facilitates the exhausts of mapping of a human organisms health snapshot into words (ii). System of querying the user so as to translate his or her responses into the categories of said taxonomy that would allow complete mapping of their health snapshot, (iii). A means of encoding information content of the user health snapshot into numerical values that can be manipulated by digital computer, and finally (iv) a method of processing the encoded information representing a user's health snapshot so as to allow the interaction of that user's health snapshot with a database of other user's health snapshots so as to generate meaningful inferences and analysis of the user's health snapshot so as to output meaningful information to the user.

SUMMARY OF THE INVENTION

A system and method are presented for the articulation, in data structures which can be operated upon by digital computers, of the health snapshot of a human being, and the interaction of that human's health snapshot with a database of other system users' health snapshots so as to obtain information and meaningful problem solving approaches with regard to the state of the human being's health. Although the techniques described can be applied to any comprehensive description of an organic or other system (e.g., horses, a chemical manufacturing system, an automobile) and any database cataloging events and problems experienced by, possessed by, or involving such systems, in the preferred embodiment the system under consideration is the mental and physical health of the human

organism, and the database of systems and their events is a collection of the comprehensive descriptions of the health of a multitude of people. Each such health snapshot, or systemic description, comprehensively describes a persons health in terms of system common categories.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will be more readily understood from a detailed description of the preferred embodiments taken in conjunction with the following figures. Many of the drawings consist of screen shots of an exemplary embodiment of the invention adapted to the World Wide Web. In this embodiment the trade name "Medigenesis" is used to denote the system, and as such, appears on many of the screenshots.

Figure 1 is a screenshot of an exemplary system homepage;

Figure 1A depicts the system structure and data flow;

Figure 1B depicts a simplified version of the system structure and data flow;

Figure 1C depicts the descending levels of abstraction of user events;

Figure 1D depicts the fields of a Patient Description Vector;

Figure 1E depicts the clustering concept;

Figure 2 is a screenshot of an exemplary "What is Medigenesis" informational page;

Figures 3-3C depict an exemplary "Your Privacy and Security" page;

Figure 4 depicts an exemplary "New Member Information" box from the account signup page;

Figure 5 depicts an exemplary "What's News" screen;

Figure 6 depicts an exemplary "Contact Us" screen;

Figure 7 depicts an exemplary "Provider Resourses" screen;

Figure 8 depicts an exemplary "Reading Room" screen;

Figure 8A depicts a fuller view of the exemplary "Reading Room" screen;

Figures 9 and 9A depict an exemplary "Discussion" screen;

Figure 10 depicts an exemplary "Glossary" screen;

Figure 11 depicts an exemplary "Help" screen;

Figures 12 and 12A depict an exemplary "Member Homepage" screen;

Figures 13 and 13A depict an exemplary "Recommended Groups" screen;

Figure 14 depicts an exemplary "Infertility > Subscribe" screen;

Figure 15 depicts an exemplary "Discussion > Subscribed Groups" screen;

Figures 16-17 depict an exemplary "Event Locator" shown for a child female user;

Figures 18-22 depict the "Event Locator", shown for an adult male user;

Figure 23 depicts an exemplary "Locate a Treatement" screen;

Figure 24 depicts an exemplary "Locate a Treatment Screen" with a list of antibiotics displayed;

Figures 25 & 25A depict an exemplary "Treatment Details" screen;

Figures 26-32 depict examples of the help screens;

Figure 33 is an exemplary depiction of

Figure 33 is an exemplary depiction of the Member Homepage;

Figure 34 is an exemplary depiction of the Your Health Profile page;

Figure 35 is an exemplary depiction of the Member Information page;

Figure 36 is an exemplary depiction of the Treatments page;

Figure 37 is an exemplary depiction of the Primary Problems interface;

Figures 38-41 are an exemplary depiction of a Medical Summary Report; and

Figure 42 is an exemplary depiction of the Diagnostic Tests page.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

You are chatting with an old friend at a party. After catching up on the latest baseball scores, your progress on a new project at work, an interesting recipe you tried recently for pasta premivera, and you mentioned you are worried about your oldest son. He is fourteen (14) years old, has developed acne, and his egama has gotten worse. He is really self-concious about his skin; one knows how children are at that age. That may account for why he has been getting such terrible stomach aches and headaches lately. The doctor wants to start him on antibiotics for the acne. You hate the though of him taking that stuff but what else can you do? "That's funny" replies your friend, as it turns out his cousin has a thirteen (13) year old daughter with strikingly similar problems; as it turns out she has developed an allergy to dairy products. Your friend continues, that simply by cutting most milk, cheese, and ice cream out of her diet her acne, eczema and stomach problems cleared up in less than one month.

Take that scenario and multiply it by thousands and thousands of people, and you have the idea of the preferred embodiment of the present invention. The system allows the user to tell it, via an automated graphical interface, about his or her medical problems, symptoms, lab test results, history, intuitive vague feelings about his or her health - - in short, all the details that make a person, medically speaking, who they are - - then automatically guides the user through a comprehensive questionnaire to take the user's comprehensive health description. Handing the acquired data to an information processing module the system then matches up the user with others within the system database that medically speaking "look just like the user", on the assumption that what has worked for them has a solid chance of working for the user.

The system of the preferred embodiment of the invention is therefore a tool of efficiency. It takes into account everything that makes the user who he or she is, mines the data inherent in the system database, and uses that interaction to generate a report that lists a variety of proposed therapies that have given other similarly situated users benefit. The system is also a tool of empowerment. It helps its users take better care of themselves and their families. After interacting with the system database, a user will be able to generate a medical profile of themselves, their child, or their parent, to share with their doctor. This allows the user to become a better informed patient of his or her doctor, thereby increasing the effiencies of physician provided therapies, as well as being able to ask the relevant questions, having been mentally prepared and informed by the system of the preferred embodiment of the invention well in advance.

In the past, medical databases were available only to medical professionals. The data they contained were in a language commonly spoken only by such professionals. By contrast, the system of the present invention uses ordinary language to interface with its users. It describes symptoms in the same words that a user might utter when talking to their doctor. It can be used by anyone, for anyone, at any time. Since it avails itself of widely accessible computer networks linking multitudes of individuals, such as the Internet, and is completely scaleable, the database can easily accommodate hundreds of thousands, or even millions, of users. The vast scale of the invention implies that there are bound to be a significant number of other users who look, medically speaking, very similar to the user. This offers him or her the benefit of the medical experiences and data of these other "medically similar" users. In effect, the system of the preferred embodiment of the invention is the largest continually operating cocktail party ever known.

However, in the case of the preferred embodiment of the invention, what the user finds transcends the best of imaginable cocktail parties. The system functions as an expert system that knows how to, most efficiently and comprehensively, query each attendee at the virtual cocktail party so as to coax them to articulate a comprehensive and complete expression of their medical state of being. Further, and at the same time, functioning like the stereotypical cocktail party gadabout, the system immediately communicates all useful information contained in the totality of the minds, bodies, and experiences of all of the other guests at the cocktail party to the user, so as to better inform and empower the user as to the state of their health and well being.

The system of the preferred embodiment of the present invention is constructed to accomplish three (3) functions. Information acquisition, information processing, and information output. Between the steps of information extraction and information processing there is an additional step of information encoding, and subsequent to the information processing step is another step of information decoding. While these coding/encoding steps are fundamental, they are simply means to interface the information between the user and the processing capability of a digital computer; in that sense they are secondary functions to the three main objectives of the preferred embodiment of the present invention.

Fig. 1B illustrates a simplified overall process flow, illustrative of these three phases. The three phases are delineated by the horizontal lines dividing the chart into three parts. The information acquisition phase comprises obtaining the User Provided Data 1B10. The information processing phase comprises (a) generating the Patient Description Vector, or PDV 1B20, which is how the system "sees" the user, and (b) the generation of the cluster of similar users 1B30 in a "medical distance" sense, where greater similarity generates a larger score. Finally, the information processing phase comprises analysis of the cluster of medically similar users and the generation of reports 1B40 to the original querying user.

The information extraction phase consists of obtaining a complete and comprehensive snapshot of the individual user's health picture. In the language of system theory, a

complete description of the system state is here elucidated. This is accomplished using the system's unique taxonomy. The taxonomy is a language or lexicon that is detailed enough so as to allow the system to store a comprehensive description of the user which facilitates finding medically meaningful similar users, and at the same time comprises language that is natural enough to allow even the uneducated and unsophisticated user to meaningfully articulate his or her own medical state of being.

The information processing functionality is a unique method of what is known in the art as data mining or knowledge discovery. It involves a two (2) step process: (i) statistical processing of the system database to locate a set of other users similar to the querying user, and (ii) analysis of the set of similar users to find hidden patterns and useful remedies, possible solutions, therapies, and information. A simple example of such remedies would be the idea avoiding of dairy products which was exchanged between the two attendees to the example cocktail party discussed above. In the system of the preferred embodiment of the invention, however, this would not be a random, anecdotal, and unquantified piece of information exchanged between people chatting at a cocktail party. Rather, a statistically significant correlation between persons in the system database similar enough to the querying user to provide meaningful health analogies.

Knowledge Discovery In The Preferred Embodiment Of The Present Invention

Before describing in detail the three stages of the system of the preferred embodiment of the invention and the detailed interactions with it which a user would undergo, it is first

necessary to understand what the actual goal or functionality of the system is. This requires some appreciation of the underlying analytical techniques that support knowledge discovery from the system database. Because the system of the preferred embodiment of the invention is interdisciplinary in nature, i.e. it touches on the areas of semantics and the creation of a linguistic version of an orthogonal basis set, system theory, medicine and healthcare, and finally, data mining, knowledge discovery, and statistical analysis, it is felt necessary to provide some general conceptual background.

Next described, therefore, is what was termed above the information processing step of the preferred embodiment of the present invention, which relates to the general discipline of statistical analysis and data mining.

Different data mining methods can be employed to provide a "microscopic view" of the data which enable the detection of invisible patters among large numbers of recorded user histories. Using an assortment of data mining techniques users will be able to have a direct "knowledge exchange" with a structured database containing records of other users, their symptoms, and what medical options have worked for them. A key knowledge extraction technique that is employed in the preferred embodiment of the present invention is cluster analysis, sometimes known in the art as proximity analysis, or nearest neighbor analysis. Cluster analysis is an exploration of a data set of vectoral representations of database members, or entities, for the identification of natural groupings. The resulting natural groupings class similar entities together, and within a group the entities share similarities in the attributes that characterize them. In such

cluster analysis no assumption is made about the number of underlying groups or any other structural aspect. Grouping is done after defining an appropriate similarity or distance measure. Typical example applications of clustering are customer segmentation and database marketing. Once the customers are divided into homogenous clusters, each cluster can be identified by cluster profiles or average cluster behavior. In the system of the present invention users are characterized in term of a representational vector, where the vector represents the user's medical situation/experiences, or what has been termed herein the "medical state of being."

As those who are skilled in the art will readily understand, this technique is sometimes referred to as nearest neighbor analysis. In nearest neighbor analysis an algorithm is constructed to find the nearest neighbors in a certain class or universe to which a given element belongs. In the system of the preferred embodiment of the present invention, not just *the* nearest neighbor is desired, but an entire set, or cluster, of nearest neighbors is desired to provide medical analogies for the query user. The set of nearest neighbors is defined by a dynamic algorithm which decides how near the set of nearest neighbors must be to the querying user in the multidimensional vectoral space which is the conceptual computing environment of the system. As will be readily obvious to those skilled in the art, one of the operands of the nearest cluster algorithm will be the "medical distance" measure assigned to the distance in the multidimensional vector space between the querying user and each of the other users in the database. This distance metric algorithm is itself dynamic and will be continually self optimizing so as to more and more optimally articulate the distance, in a meaningful medical/health sense (measured as the capability

to provide useful treatment or diagnostic analogies and guidance) between any two users in the system database.

Another data mining technique that is often employed is the discovery of association rules. Association rules discover the correlations between attributes, such as, the presence of one particular attribute implying the presence of other attributes for an entity. An example of an association rule is that “whenever a given customer purchases salmon and mussels he also buys white wine”. In commercial contexts, association rules are often used in cross marketing, store layout planning, catalog design, and the like. For two (2) sets of items x and y, an association rule is usually denoted as $x\sim y$ to convey that the presence of the attribute x in a transaction implies the presence of y. The role of associations would be complementary to clustering (once the clusters are determined, mining for association rules within the cluster would provide useful information on the medical experiences of the cluster members).

These two primary techniques, clustering analysis and association rule discovery, are further extended in the system of the preferred embodiment of the present invention to include classification approaches, where real time classifiers are run to answer user posed questions. Classification deals with sorting a given set of observations into two (2) or more classes. The emphasis is on deriving a rule that can be used to assign a new observation to one of the classes, i.e., future predication. A classic example of classification is depiction of a disease. A classifier can be calibrated using a data set containing disease present and non-present vectors. Then it can be used to predict

whether new patient vectors have the disease or not. Another example, from recent medical literature in the area of autism, is the detection of an environmental factor or factors significantly increasing the risk of autism. As is well known in the medical community dealing with autism, there has been established, in a statistically significant sense, a connection between children receiving the combined MMR vaccine (mumps, measles, and rubella) and the incidence of autism. Thus, a classifier could then be calibrated using a data set from the system database of autistic children containing those who received the combined MMR vaccine and those that did not. Then the classifier can be used to predict whether new users who received the combined MMR vaccine have, or, have a risk of developing, the disease or not.

Fig. 1A depicts the data flow in the preferred embodiment of the invention. Beginning with the User Reported Data 1A01, a user logs on to the site, and via an anatomical user interface and a comprehensive questionnaire, as described below in connection with the user interface, reports all relevant data to his health snapshot. Conceptually, this data allows the system to comprehensively describe the user's health "system" (to analogize to system theory), or her comprehensive medical state of being. This report is in the language of, and is stored in, the system databases allocated to each user, as a series of User Reported Problems/Events. How this information is elicited from the user is fully described below in connection with the user interface, and relates to the information acquisition aspect of the preferred embodiment.

Exhibit A-1 is an example listing of all user reportable or identifiable Problems/Events that are possible in the preferred embodiment, entitled EVENT LOCATOR. This list is

dynamic, however, and can be modified as warranted by the continual internal system monitoring, for efficiency, clarity and comprehensiveness. As its name implies, the listing is oriented towards the Anatomical User Interface and the Questionnaire, as described below, and thus is organized first by the anatomical location on the body where the problem or attribute is manifested. This listing, having some 32,000 possible ailments or attributes, is simply too large to be used to represent the user in the system. Thus, it must be collapsed into more general groupings. Exhibit A-2, entitled "Medex_Formal Problem", is such an example distillation. This Exhibit has three columns. The middle column, MEDEXNAME, contains 5,597 unique user events, to which the entire 32,000 symptom aliases can be mapped. The third column (rightmost) describes whether the event is a medical problem, such as, for example, a spine injury or an allergy to latex, or simply a pertinent medical fact, termed an "attribute", such as, for example, having had a certain standard vaccine, or having traveled to a particular foreign country. The first (leftmost) column of Exhibit A-2 is the SFWID, which is an example set of 2204 possible System Function Where ("SFW") combinations. The SFWs, as more fully explained below, are the orthogonal categories by which a user is comprehensively represented in the system. Fig. 1C illustrates the increasing level of abstraction (going down the page) moving from the circa 32,000 symptom aliases to the circa 5600 problem names to the some 2200 SFWs.

Obviously there are two levels of abstraction ending up at the same place – the 2200 SFWs. Why? One purpose of the symptom alias is that it provides for members to describe a specific problem 'in their own words'. The example that always seems to get

used to demonstrate this was 'stinky poop' versus 'smelly feces' versus wickedly pungent excrement. All say the same thing, yet each uses different words reflective of the user's soci-economic stratum and linguistic habits. Thus the some 32,000 symptom aliases have significant synonymy and semantic redundance.

The other reason for duplication is that a symptom can appear, as shown below, in more than one place in the event locator - a person may click on arm, then skin, then 'eczema on the arm', or they may click on skin, then 'eczema on the arm'.

SFW – System Function Where:

the Central Data Structure of the System

SFWs are organized not by location (visually perceived spatial orientation), but much more efficiently by bodily system and function (conceptually perceived functionality), the latter being the reported problem or condition. The lowest level of abstraction of the SFW is the Where element, and identifies where anatomically that particular system's particular ailment or condition is manifest.

Exhibit A-3 contains an example listing of a set of 2204 SFWs, comprising an orthogonal basis set of medical conditions and facts by which a user's health state of being can be thoroughly expressed. The information processing module of the system of the preferred embodiment sees each user as a vector comprising an age component, a gender component, and N SFW components, where N is the number of all SFWs possible in the

system. In the example listing of Exhibit A-3, N = 2204. Fig. 1B depicts the increasing levels of abstraction between Exhibits A-1, A-2, and A-3.

Because an individual user may report data a number of times, but is represented by only one data structure within the system, multiple occurrences of a user event are collapsed into one value for that particular SFW, using an equation that maps one value to the SFW in question, including information regarding the number of occurrences and the severity of each occurrence. Referring again to Fig. 1, the reported information by the user 1A01, and the severity parameters 1A02, are distilled and combined to create the Patient Description Vector, or "PDV", which, as described above, is how the user is "seen" by the system's information processor.

An example of an SFW component of the PDV encoding the fact that a user has Eczema on the arm would be, in the example of Exhibits A, coded as "Skin-Inflammation-Not Specified" as is shown on the top record of page 110 of Exhibit A-2. Similarly, every member problem (or, synonymously, member event) from Exhibit A-1 has a corresponding SFW in Exhibit A-3.

PDV – Patient Description Vector

The PDV is a row of numbers that collectively define the point the relevant user occupies in the multi-dimensional hyperspace of all possible (considered) medical conditions. Each column in the row corresponds to a dimension in the hyperspace, and columns will be set

aside for the following pieces of information, with reference to Fig. 1D: user's age, gender, and a column for each valid SFW.

Column 1: Gender=Male and Column 2: Gender=Female

The members' gender information will be encoded by placing a 1 in the appropriate column. No information (equivalently, a zero) will be placed in the other column.

Columns 3-17: Age

The age information will be encoded by placing a 1 in the appropriate column, and zero in the other columns. Each column will represent an age range of 7 years. So, if the member is younger than 7 a 1 will be placed in the first column, if they are younger than 14 (but older than 7) a 1 will be placed in the second column, etc.

Columns 18-2221: SFWs

As per Exhibit A-3, in an example of the preferred embodiment there are 2204 different, valid combinations of SFWs. Each of these will be assigned an identification number (an 'SFWid'), and each SFWid will in turn be assigned a 'column' in the PDV vector. Thus, the total columns in the vector in such an example are 2221, 17 for storage of the age and gender information, and 2204 for the SFWs.

The value which will be placed in the column corresponding to a given sfwid is given by the following equation.

$$PDV_{sfwid} = 1 + U_{pperB} \cdot \left[1 - \left(\frac{1}{a}\right)^{i} \left(\frac{1}{b}\right)^{j} \left(\frac{1}{c}\right)^{k} \left(\frac{1}{d}\right)^{l} \right]$$

Where:

PDV_{sfwid} = The number to be placed into the PDV vector for this sfwid. The parameters allow multiple occurrence information, as well as severity information (since there is no separate SFW for a severe, mild, or medium occurrence of the same event) to be encoded in the SFW value. These parameters operate as follows:

- U_{pperB} this parameter bounds the maximum that can be reached in an entry. (The actual maximum that can be reached is 1+UpperB);

- **a** – parameter that controls the rate at which each extra '**mild**' event (classified within this particular sfwid) brings the entry towards the upper bound;

- **b** – parameter that controls the rate at which each extra '**moderate**' or '**variable**' event (classified within this particular sfwid) brings the entry towards the upper bound;

- **c** – parameter that controls the rate at which each extra '**severe**' event (classified within this particular sfwid) brings the entry towards the upper bound;

- **d** – parameter that controls the rate at which each extra '**variable**' event (classified within this particular sfwid) brings the entry towards the upper bound;

Input numbers (dependent on user data)

- **i, j, k, l** – these numbers count the number of (respectively) mild, moderate, severe, and variable events that the given member has had, or currently has, which are classified to fall within this **sfwid**.

These severity parameters (which include the multiple occurrence information) are shown as an operand to the PDV in Fig. 1A, item 1A02.

These equations operating on the user provided data will lead to the generation of a vector 1A05 with reference to Fig. 1, where the number of columns, or the dimensionality of the hyperspace (n), will be on the order of 2200. Basically the PDV is simply a format to describe the member in a way conducive to 'proximity analysis'. Once the PDV is generated in the above fashion, it will be stored in the database for later retrieval, and for usage in reporting / debugging purposes.

Metric Calculation

Having To find the similarity between two members (as represented by their respective PDVs) a 'metric calculation' is undertaken. This metric operates as a variation on the dot product (which is a scalar measure of the extent that one vector lies along the direction of another, itself a measure of similarity; the dot product of a vector with itself is thus unity). The metric can be weighted to take into account that the dimensions, being word based and subject to interpretation, may not be absolutely orthogonal, or independent, and thus the coincidence of two different SFWs may actually deserve a significant similarity rating.

Calculation of the Metric

A crucial part of the system is the calculation of the 'similarity' between two PDV vectors. This step is shown as 1A10 in Fig. 1A. In the preferred embodiment, the formula used to calculate the 'similarity' between two PDV vectors, **x**, and **y** is given by:

$$similarity\_measure = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} w_{ij} \cdot x_i \cdot y_j \qquad \left(\forall_{i,j} : w_{ij} > \tau\right)$$

Basically, the system multiplies every non-zero entry in **x** against every non-zero entry in **y**, using the corresponding component of the appropriate weighting factor matrix **W**, 1A06. The system then sums the result, completing the medical distance calculation 1A10.

However, where the weighting term (w) is zero, or when w is less than some (adjustable) threshold tau, that term is not counted in the summation, and no similarity is credited for the coincidence of the two SFW fields involved.

The above medical similarity metric 1A11 is actually a variation, or extension, of the well known 'dot product'. Obviously, it is dynamic, and can be easily changed so as to optimize the meaningfulness and usefulness of the medical similarity concept.
The calculation of the metric can be understood, by considering, first, the 'dot product'. If we have two vectors in an n-space (in 2-space we might consider the closeness between

two directions, or between two 2-D vectors), the simple dot product of those two vectors, **x**, and **y**, is given by:

$$= \sum_{i=0}^{n-1} x_i \cdot y_i$$

In the case that W in the metric discussed above had all ones in the diagonal, then the metric reduces to a normal dot product. That is, if

$$W = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & \dots \\ 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & 1 \end{bmatrix}$$

then the metric is simply a straightforward dot product.

The way that the similarity metric calculation works can be adjusted by adjusting the parameters in **W**. It can also be adjusted, more easily, by changing the threshold tau.

Finding the cluster

By repeated application of the metric (or some optimized equivalent) it will be possible to find the n members who are 'closest' to the current member 1A15. This list of 'cluster buddies' (having the highest scores on the similarity metric) will then be stored temporarily, for use in subsequent calculations.

Storage and retrieval

The system of the preferred embodiment supports the storage and retrieval of data relating to cluster analysis. PDV information in particular, being multiple thousands of columns wide, needs to be stored in a data-compressed way, and yet, be retrievable in a vector format. The primary data stores include the following.

- PDVs (the latest PDV for every member, including the calculated length of that PDV);
- Similarity matrix information (the matrix of similarities calculated between PDV's, or equivalently, members, being of size M x M,where M is the number of members, or alternatively, a vector of 1 x M for each member, stroing his or her similarity measure from all of the others); and
- Supporting information for the metric calculation (the weighting matrix **W**).

Performance

A good part of the accuracy for this method of measuring 'similarity' between members depends on the exact values chosen for the weights matrix W, and for the threshold Tau. A high threshold (or a lot of zeros in W) leads to **less dimensionality** in the calculation

and consequently more tractability in trying to find similar members. On the other hand, a low threshold tau (or a lot of high numbers in **W**) is equivalent to saying that all factors in the body are tightly interrelated, and consequently a high dimensionality in the calculation. The same trade off applies to the question of whether the UpperBound, and the a, b, c, d parameters are set high or low during the generation of the PDV. Therefore, it is expected that the method for generating **W**, and the choice of optimum values for the other parameters will evolve to higher precision and better predicatability. The method of the preferred embodiment for achieveing this evolution is to define one or more success measures, and create a genetic algorithm to automatically periodically diagnose system performance in terms of the one or more success measures, and automatically modify the various equations for the similarity metric, for **W**, and for the severity and multiple occurrence parameters.

Determining The Optimal Number Of People To Display (Creating Dynamic Clusters)

In basic applications of clustering, groups (or clusters) are formed a priori in the metric space, and a new individual is mapped to the closest group. In the approach of the preferred embodiment, the distance (metric) of the new person to each of the persons (historical) in the database is calculated, and we select people that are "close" to him in a ranked manner. In such a scheme, the question arises: How many people are "close enough" to the new person? One logic would be to show the people that closer than a certain threshold (these people would then be showed in a ranked manner, closest to the

farthest). Similarly, a certain fixed number of cluster buddies or a percentage of the total number of database members could be chosen. Optimally, it is desirable to let the data itself determine the natural boundary. Other users in the vicinity are included in the cluster until a gap is encountered that is bigger than a gap threshold. The logic can be visualized in the plot shown in Fig. 1E:

In Figure 1E, the points lying in the Region A are considered close, and the points lying outside region B are considered "not close".

Consider the following series of distances:

Distances: 1, 1.2, 2, 2.5, 3.0, 4.0, 7.0, 8.0, 8.5

The gaps between successive prospective "cluster buddies" are then:

Gaps: 0.2, 0.8, 0.5, 0.5, 1.0, 3.0, 1.0, 0.5

Gap Moving Averages:

Moving Average 1 = 0.2;

Moving Average 2 = (0.2 + 0.8)/2 = 0.5

Moving Average 3 = (0.2 + 0.8 + 0.5)/3 = 0.5

Moving Average 4 = (0.2 + 0.8 + 0.5 + 0.5)/4 = 0.5

Moving Average 5 = (0.2 + 0.8 + 0.5 + 0.5 + 1.0)/5 = 0.6

At this stage, the next gap (=3.0) is significantly greater (order of magnitude = 5) than the current gap moving average of 0.6. Hence the point may not be desired to be included in the group, and the cluster is restricted to the first 5 cluster buddies.

User Interface and Data Acquisition

What has been described above relates to the information processing aspect of the preferred embodiment of the invention. Temporally, this information processing stage occurs after the information acquisition stage, where the complete systemic description of a user's medical/health state of being is elicited, and mapped to the SFW's comprising the Patient Description Vector, PDV. What will next be described is the information acquisition aspect of the preferred embodiment.

The system of the preferred embodiment of the present invention is implemented on a computer network, such as the Internet. The user's gateway to the system is the Home Page, as shown in Fig.1. Clicking on button 101 leads to a mission statement page, as shown in Fig.2. Clicking on button 102 leads to the Your Privacy and Security page, shown in Figs.3-3C.

Clicking on button 104 accesses the Account Signup page, as shown in Fig.4, and the New Member Information box appears as therein depicted. The user fills out the interactive box and receivesaccessto the site. Button 105 leads to the What's News? page, as shown in Fig.5, and button 106 leads to the Contact Us page, as shown in Fig.6. Finally, button 107 leads to the Provider Resources page and subpages, as shown in Figs.7-7B. The menu bar, which is always at the bottom of the system screen, wherever one is in the system, will now be described, still with reference to Fig.1. Menu Item 108 leads to the Reading Room, as shown in Figs.8 and 8A, Item 109 leads to the Discussion Area, as shown in Figs.9 and 9A. Item 110 leads to the Glossary, depicted in Fig.10. Recall that one of the functions of the site and the system is to educate the user in the terms used to describe his or her health, so the glossary is quite an important tool. Item 111, Help, displays the help informational screen as shown in Figs. 26 – 32.

Item 112 leads the user to a system search screen.

The critical interactions between the user and the system of the invention occur in the information acquisition phase, which occurs when the user, interacting with the system interface, describes his detailed state of health, the treatments he is taking, his primary and secondary problems, and the results of lab tests.

The interface operates as follows. From the system Home page, shown in Fig.1, upon clicking on the Member Home Page button 113, the user is taken to the Member Home Page, and sees the screen depicted in Fig. 33.

Upon clicking on the Your Health Profile button 3301, or the "go" sign to the right of it, the user is taken to the Your Health Profile page, and sees the screen depicted in Fig. 34.

There are six categories of information which can be entered and managed (i.e., edited) by the user at this page. Member Information, Treatments, Primary Problems, Secondary Problems, and Diagnostic Tests. The Medical Summary category cannot be edited, inasmuch as it represents the output from the system to the user, or for the benefit of the user's physician, but new summaries can be run by the user at any time, and are intended tobe run if any of the other data has changed. Clicking Member Information 3401 takes the user to that page, and displays the screen depicted in Fig. 35.

At this juncture the user can modify or add to any desired information that has already been stored, and then click at the button labeled Return to: Your Health Profile to return to the Your Health Profile page.

With regard to Treatments, listed as the second category on the Your Health Profile page, Clicking on Add Treatment brings the user to the Add Treatment page, and the text and interactive box appears as depicted in Fig. 36:

The function of this screen is for the user to tell the system database which treatments, meaning primarily medications, that he or she is currently taking. This information is necessary to obtain the true picture of the user's health. With reference to Fig. 23, the user sees the Locate a Treatment interactive box, and can either search for a treatment by typing in a text string in the type-in box 2305, or choose a treatment category 2306, by clicking the menu selector 2307, and clicking on the list button 2304. The latter action will bring up the Health Option List for the selected type, as in Fig. 24, where a list 2403 of the chosen type, here antibiotics, is shown. Clicking on a particular listed treatment, such as, for example, the antibiotic Zyvox 2402, brings the user to the treatment details screen.

Figs.25-25B also depict this screen. Here, with reference to Fig.25A, the user discloses the date the user stopped taking the medication 25AO1,the good response descriptor 25A02, or the bad response descriptor 25AO3,comments for either good or bad responses, 2SAOS and 2SAO6,respectively, whether the treatment should be displayed on the progress report 25AO4 and any further comments 25AO7.The information is saved by clicking on "Save" 25AO8.The response descriptors for good and bad are shown in Fig.25B,in box 25BO1, and

range from mildly bad (good), somewhat bad (good), bad (good), to seriously bad (good). After completing the information for the treatment, the screen depicted in Fig 37 is next seen.

The user either adds a new treatment and repeats the process just described, or continues with the health profile of the six information categories found at the "Your Health Profile" page, the most important are the Primary Problems and the Secondary Problems. These will be next described in detail.

From the Your Health Profile screen a user accesses the primary problems screen by either clicking on the Add Primary Problem or the manage primary problem links. This takes the user to the Event Locator, as shown in Figs. 16 and 17, for a young female child, and in Figs. 18-22 for an adult male. The user clicks on the body of the Event Locator Figure 1801 in Fig. 18, and a part of the body is highlighted. Alternatively, the user clicks on one of the words located around the figure. In either case the chosen body part or topic appears at the top 1805 of the interactive box on the right of the screen, and a list of "aliases" or sub categories of the chosen category appear for choosing and adding to the problem list shown in the Chosen Problem List box 1802. The user continues in this fashion until all the primary problems are chosen. The user then returns to the Your Health Profile page by clicking on the save and return button 1806, and sees the modified Primary Problems section as depicted in Fig. 37.

Secondary Problems are queried by an exhaustive questionnaire. Sample pages of the questionnaire are provided as Exhibit B-l. The questions seek to elicit the various problems a user has, and track the Exhibit A-1 set of all possible problems in all possible phrasings inherent in the system.. As described above, the critical information gleaned is mapped to the SFWs and stored in the user's PDV.

Clicking on the "Run a new Medical Summary Report" link from section 5 of the Your Health Profile page generates a report, an example of which is shown in Figs. 38-41. With reference to Fig. 1A, this is step 1A20. The report, *inter alia,* is characterized by an informational display similar to the following example text:

Your Cluster[1]

Number of people in your cluster: 23
Defining symptoms in your cluster:

Within your cluster, the following percentage of people have experienced symptoms exactly like, or similar to your problems…

	Exact[2]	**Similar**[3]	**0% … 100%**
>> headaches[4]	10%	30%	*(Bar Chart)*

---

[1] *This information is presented as part of the model report.*

[2] *This value indicates the percentage of members (in the cluster) having at least one event with the* ***exact*** *same formal problem id.*

[3] *This indicates the percentage of members (in the cluster) having* ***at least one*** *event which is 'similar' to the event listed. Here similarity is defined as a match on the SFW record.*

>> staph	30%	40%	*(Bar Chart)*

Other people in your cluster also had...

>> allergy to gluten[5]	70%	*(Bar Chart)*
>> red hair	60%	*(Bar Chart)*

Treatments in your cluster

People within your cluster have reported good responses to...

>> magnesium	70%[6]	*(Bar Chart)*

People within your cluster have reported bad responses from...

>> trepanning	20%	*(Bar Chart)*

Common discussion forums for people in your cluster

If you wish to share information, or collaborate with others who are 'like you' then you will be interested to know the forums they are subscribed to:

>> staph infections forum	70%	*(Bar Chart)* *<<subscribe>>* [7]

---

[4] *The members chosen alias (as chosen on the Event Locator, Exhibit A-2, for example) is used to label the events listed in the rows here.*

[5] Options listed here are those with more than one exact match (in the cluster) on formal problem id (but not shared by the member in question) - consequently, the formal problem name is used to label the events listed here.

[6] *This gives the percentage of people **within the cluster** who have had good responses to this treatment. (Not the percentage of people who have taken the treatment who indicated a good response)*

[7] *Clicking the 'subscribe' link will take you to the default interface for subscription to a group.*

>> headaches forum	20%	*(Bar Chart)* *Already subscribed*

Note that the report summarizes the reported problems, provides the benefit of the system's statistical analysis, and can even suggest, based upon such analysis, further diagnostic tests. As well, the report draws on all the information stored in the system, and not just that information encoded in the PDV. Thus, if the user complies with the suggested diagnostic tests, assumably she will report the results of the diagnostic test to the system, generate a new medical summary report, and both she, and the knowledge inherent in the system, will obtain further useful information. Clicking on the Manage Diagnostic Tests link at section 6 of the Your Health Profile Page displays the screen shown in Fig. 42.

The system thus serves as the direct recipient of laboratory tests, and reports the results back to the user. Clicking on the link 4201 at the top right, or using the go button 4203 and menu bar 4204 returns the user to the Your Health Profile page.

To use a signal processing analogy, the bandwidth of the information acquired in the information acquisition phase is simply too great to be processed in real time by the information processor. Thus, for the purposes of generating a cluster, the signal is downsampled, and high frequency information is discarded. Once, however, the cluster is found, and computation does not require all the users in the system database to be operands to the processing algorithms, the bandwidth can again be increased to the original bandwidth, and all information, no mater how complex, available in the system regarding the user, **and the other members of the cluster**, is available for analysis in generating the user reports. With

reference to Fig. 1A, the cluster 1A15, and all of its users' complete records, as well as the user's complete original records, collectively 1A16, are available as operands to the report generating algorithms.

Thus, once the cluster closest to the new user is arrived at, additional analysis such as data mining using association rules is employed to derive useful information for the nearest users, as above. One of the data mining techniques employed is the discovery of association rules. Association rules discover the correlations between attributes, such as the presence of a particular attribute implying the presence of other attributes for a user. As described above, for the sake of analytical tractability, many auxiliary dimensions, elicited in the user interface from the user, but not encoded in the SFWs, were omitted from the original clustering. These dimensions, such as aggravating factors, alleviating factors, etc. (see Exhibits A-1 and A-2) hold rich information that has, in the SFW encoding and cluster generation process, been unexplored.

An example of an association rule is that "whenever a patient has disease X, the common aggravating factor is wheat". For two sets of items X and Y, an association rule is usually denoted as x~y to convey that the presence of the attribute X in a vector implies the presence of Y. The role of associations would be complementary to clustering (once the clusters are determined, mining for association rules within the cluster provides useful information on the medical experiences of the clusters).

Primary Scenario

To summarize the operation of the system of the preferred embodiment,the flow of events, in the usual case, is as follows.

1. A member accesses the system, and completes the steps in the Your Health section. (Detailing their Primary Problems, Treatments, and taking the Questionnaire, all as described above).

2. The User (Member) chooses to generate a new Report.

3. The original User's record is mapped to a PDV, based on the medical information that the user has entered. This discards some information in the User's record for the purposes of generating the cluster.

4. The PDV, and supporting user choices from the Exhibit A-1 list, as well as the formal problems of the A-2 list that the A-1 list choices are mapped to, is stored for later retrieval.

5. The PDV is compared against all existing PDVs in the database to find a cluster of members (users) who are 'close' to this member.

6. Queries are generated against the top 'n' members to determine their most common discussion groups, defining problems and good/bad treatments. All available information in the system is used at this stage.

7. This information is presented to the user in a table, or other meaningful and efficient formats.

8. Reports can be sent electronically, or via hard copy, to a User's doctor or other designated parties. Fig. 1A30.

Event Locator and Questionnaire Design Issues:

The design issues behind, and the functionalities of, the Questionnaire, will next be discussed.

The capacity of databases to permit new methods of viewing patterns of information and finding matches is not worth much without ways to capture accurate, detailed, and structured input.

The user is the original, most reliable and most efficient source of most information about symptoms, life events, environmental exposures, past illness, operations, allergies, and family history. The user has a story - referred to medically as the medical, social, environmental, family history. The system database has rows and columns waiting to

receive the story. The interface between the input and storage of this data fulfills the following criteria:

1. Engaging;
2. Intuitive;
3. Uses everyday language;
4. Codes the data on entry.

Questionnaires in current medical use have narrow or superficial areas of interest in information that can expand in the context of a personal interview. There does not now exist a method for the free-form capture of detailed coded data in a system that begins with the same kind of question one would ask when sitting down with a patient for the first time: "Please tell me what is bothering you?" The Event Locator (Figs. 16-22, and the listings in Exhibit A-1 which can all be addressed in the Event Locator and/or follow up Questionnaire) starts from that point and leads to a questionnaire that follows up on symptoms and other events captured in the event locator, as described above.

A database providing vernacular descriptions of most medical symptoms and events matched to their coded dimensional meanings provides the foundation for the preferred embodiment of the present invention's capacity to encode natural language descriptions.

The present invention's first device is a graphic representation of a figure corresponding to the user's gender and age group (adult, child, toddler). The screen presented to the user shows the figure on the left. See Figs. 16-22. Moving a mouse over the figure, the user sees the names of various body areas or organs pop up in text boxes (leg, liver, intestines, nose, face, etc) and a mouse click then gives the user a list in one of the three boxes on

the right side of the screen the top 15 symptoms associated with that area (precisely, it is the upper left hand box on the right half of the screen, labeled "areas").

The user finds that selecting a small area (e.g. nose) will produce a list of problems whose associations are restricted to the nose, whereas selection of face will beget a list that includes nose problems along with eyes, mouth, chin, lips, etc. A substantial subset of symptoms can be addressed simply by reference to a part of or place on the body. Other problems may be identified by identifying the function (e.g. pain, itching) or the cause (allergy, trauma) of the symptom to be described. Thus a user with a headache may click on "pain,' or "head" to reach a list from which his or her type of headache can be selected. A person with itching on the elbows and knees may select "itching" or click first on elbows and then knees bringing them sequentially to elbow itching and knee itching.

All possible primary events in the invention's database can be found by at least one, and usually several redundant clicking choices. A primary event is one that is susceptible of being considered as a problem that would be described in response to the question, "Please tell me what is bothering you" and which would then populate the primary problem list. Thus the user can locate all sorts of trauma, allergies, pains, itching, and other disturbances of function as well as important toxic exposures and life events. Linkage of all the symptoms is assured by a table maintained in the database denoting which symptoms are grouped under subgroups (e.g., nose) and bigger groups (e.g., face). The following options appear after the top 15 symptoms (associated with the user designated area, function or trauma) list appears on the right side of the screen:

1. The user may select a symptom from the list.

2. The user may expand the list to include all the choices (i.e. beyond the top 15) in a scrollable enlargement of the top 15 list.

3. The user may compact the symptom list by clicking to its left, on the human figure, on a location (e.g. nose, ear, mouth) representing a narrowing of the choices in a bigger groups (e.g. face). Similarly, for say, Life Events, the user may narrow its list by choosing the type of Life Event he or she wishes to select as a primary problem (death, job change, family change).

The user adds a problem to his or her primary problem list by clicking on the words that best describe one of his or her difficulties. The process may be repeated until the user has described all symptoms and events.

Once the graphic device has permitted the capture of the free form aspect of a medical interview in which the top of the user's problem list can be obtained thanks to the users incentive to input his or her main problems the user moves to the primary problem list screen for rating (assigning a numerical value representing the relative importance of each problem to the user), scoring (indicating whether the symptom is mildr moderate severe, or variable in its intensity) and describing (with drop down table choices) the onset, frequency, and episodic duration (when you get the headache how long does that episode last?) of each problem.

After the primary problems have been dealt with, the system moves the user on to describing her secondary problems. As described above, this occurs via the medium of the questionnaire.

The Questionnaire

The questionnaire allows for an inventory of other remaining difficulties that add detail to the sketch of primary problems and thus results in a true portrait of the user's unique combination of symptoms (events) stored in a manner that allows it to be matched with other individuals in the database as they are represented by statistical clusters. The key to the questionnaire is its presentation of branching, from general questions such as "Do you have any muscles spasms, tics, cramps, or tension?" to a specific list of symptoms that fall naturally into such a group. Questionnaire logic that recognizes symptoms entered in the primary problem list acknowledges previous answers ("We see that you have problems with headache; please tell us more about the factors influencing your headaches"), or builds from previous responses: ("We see that you have itchy elbows, please tell us if you have other itches that are important.")

The lexicon or taxonomy referred to above, i.e. the listings of Exhibit A-1 is the foundation of the questionnaire. The lexicon gives the invention the capacity to exchange information with users in a language that is at the same time vernacular, yet coded in ways that preserve the detailed individuality of each user. Unlike a paper questionnaire, in which the device of e.g., "If 'no' skip to question 161", has obvious limitations to one level of logical branching, an Internet or other data network accessed questionnaire has the capacity for many layers of branching that permit drilling down from a very general question. For example, from the general question "Do you have any skin problems or changes of any kind in your skin?" to (if yes) a group of more specific header questions which (if yes) permit the presentation of very specific skin symptoms. The more specific skin header questions have been formulated so that the vernacular terms used reflect the realities of medical dialog while their clusterings within each header question reflect

functional (pain, itching, disruption, dryness) distinctions allowing for the specific questions at the third layer of branching to be of the same general type.

The questions found in an example questionnaire cover all of the issues contained in the Exhibit A-1 listing. The preferred embodiment has approximately 7400 of them. Primary Problem categories are asked to nearly everyone, termed "header questions", and specific follow ups only to those indicating the presence of the problem. In this manner, the system "drills down" from the general to the specific, and thus hones in with great detail on the user's particular problems. Exhibit B-1 contains sample pages from the on screen version of an example questionnaire as seen by a user, depicting the skin header (or general) questions.

The skin header questions (Exhibit B-1), show how a complete inventory of skin questions was built from the lexicon by grouping words commonly expressed by patients to describe related problems.

Muscular problems provide another example of the way that the data in the database generates the terms used in the questionnaire. The question: "Do you have any tics, cramps, twitches, spasms, or muscle tension?" is a concatenation of terms joined by the functional pathology having to do with an abnormal increase in the normal function of muscles, to contract. It would not, however, due to ask a patient "Do you have an abnormal increase in the tendency of your muscles to contract?", because that description is too far from the vernacular. On the other hand, to design a questionnaire entirely on the basis of being able to think up all the variations of how people express such categories of symptoms without reference to a lexicon of how they actually did so would be impossibly tedious. With each question the user is presented with the appropriate modifiers of severity, onset, frequency, episodic duration, and overall duration (for

problems that ended in the past).

After completing the questionnaire, the user may promote problems uncovered in the questionnaire process to be primary problems if he or she appreciates during the questionnaire process that such and such a problem is, in fact, of sufficient concern to be rated among the ones that he or she mentioned in the primary problem phase (Event Locator, Figs. 16-22).

Coding Examples

In what follows, examples of possible coding are presented to illustrate one implementation of key system computational functionalities. Numerous variations are obviously possible, and the following examples are for illustration only, and in no way are intended to limit or restrict the multiplicity of possible embodiments of the invention covered by the claims.

The key steps of the preferred embodiment are:

1- Calculate the weightings matrix **W**;

2- Generate a PDV for a particular member;

3- Calculate medical similarity of this PDV to the other members; and

4- Find the cluster of nearest N members (dynamic calculation based upon moving averages not shown; considered a trivial extension of the example depicted given the discussion in the specification above).

```
----
-1- Calculate weightings matrix
----
```

This is done as a two step process:

Firstly the following code runs as a stored procedure and creates the 'first pass' approximation for the most common cases. Basically it gives a weighting of 1 if only s,f, or w are shared between two columns. 2 if two things are shared and 3 if all three are shared (ie they are the same column).

<CODE>

```
insert into clusterweightings
 select c1.clusterColumnId, c2.clusterColumnId,
  case
  when (sfw1.systemId = sfw2.systemId and
       sfw1.functionId = sfw2.functionId and
       sfw1.whereId = sfw2.whereId)
  then 3
  when ((sfw1.systemId = sfw2.systemId and
        sfw1.functionId = sfw2.functionId) or
       (sfw1.functionId = sfw2.functionId and
        sfw1.whereId = sfw2.whereId) or
       (sfw1.systemId = sfw2.systemId and
        sfw1.whereId = sfw2.whereId))
  then 2
  else 1
 end
from (clusterColumn as c1 inner join sfw as sfw1 on c1.sfwid
= sfw1.sfwId)
cross join
 (clusterColumn as c2 inner join sfw as sfw2 on c2.sfwid =
sfw2.sfwId)
  where sfw1.systemId = sfw2.systemId or
  sfw1.functionId = sfw2.functionId or
  sfw1.whereId = sfw2.whereId
```

</CODE>

Then, to refine the weightings matrix we pass over the columns again using VB code, the purpose of which is to deal with the situation that different Systems, or Functions , e.g. CNS and Behaviour are actually somewhat related, and should have some "closeness" score.

updateSystemSFW("X", "X", 1) for all columns that have the same system, i.e. "X" in two different columns.

The last stage downgrades the weight (by 1) when the where value that is shared is "not specified" (as opposed to e.g. "leg").

<CODE>

```
Call updateSystemSFW("CNS", "Behavior", 0.8)
Call updateSystemSFW("Craving", "Behavior", 0.6)
Call updateSystemSFW("Development", "Behavior", 0.6)
Call updateSystemSFW("Emotion", "Behavior", 0.8)
Call updateSystemSFW("Neuromuscular", "Behavior", 0.2)
Call updateSystemSFW("Speech", "Behavior", 0.4)
Call updateSystemSFW("Vascular", "Blood", 0.2)
Call updateSystemSFW("Metabolic", "Blood chemistry", 0.4)
Call updateSystemSFW("Digestive", "Body weight", 0.4)
Call updateSystemSFW("Metabolic", "Body weight", 0.4)
Call updateSystemSFW("Nutrition", "Body weight", 0.2)
Call updateSystemSFW("Vascular", "Cardiovascular", 0.6)
Call updateSystemSFW("Development", "CNS", 0.4)
Call updateSystemSFW("Emotion", "CNS", 0.6)
Call updateSystemSFW("Hearing", "CNS", 0.2)
Call updateSystemSFW("Immune", "CNS", 0.4)
Call updateSystemSFW("Neuromuscular", "CNS", 0.2)
Call updateSystemSFW("Speech", "CNS", 0.4)
Call updateSystemSFW("Vision", "CNS", 0.2)
Call updateSystemSFW("Eating", "Craving", 0.8)
Call updateSystemSFW("Emotion", "Craving", 0.4)
Call updateSystemSFW("Metabolic", "Craving", 0.2)
Call updateSystemSFW("Nutrition", "Craving", 0.6)
Call updateSystemSFW("Life Event", "Development", 0.2)
Call updateSystemSFW("Eating", "Digestive", 0.8)
Call updateSystemSFW("Exocrine", "Digestive", 0.2)
Call updateSystemSFW("Immune", "Digestive", 0.4)
Call updateSystemSFW("Nutrition", "Digestive", 0.6)
Call updateSystemSFW("Emotion", "Eating", 0.2)
Call updateSystemSFW("Nutrition", "Eating", 0.8)
Call updateSystemSFW("Metabolic", "Endocrine", 0.6)
Call updateSystemSFW("Reproductive", "Endocrine", 0.6)
Call updateSystemSFW("Metabolic", "Energy", 0.6)
Call updateSystemSFW("Warmth", "Energy", 0.4)
Call updateSystemSFW("Skin", "Hair", 0.6)
Call updateSystemSFW("Immune/lymph", "Immune", 1)
Call updateSystemSFW("Warmth", "Metabolic", 0.4)
Call updateSystemSFW("Skin", "Nails", 0.6)
Call updateSystemSFW("Skeletal-joint", "Neuromuscular", 0.2)
Call updateFunctionSFW("Abnormal color", "Abnormal", 1)
Call updateFunctionSFW("Abnormal growth", "Abnormal", 1)
Call updateFunctionSFW("Abnormal lab test", "Abnormal", 1)
Call updateFunctionSFW("Abnormal odor", "Abnormal", 1)
Call updateFunctionSFW("Abnormal PE", "Abnormal", 1)
Call updateFunctionSFW("Abnormal rhythm", "Abnormal", 1)
```

```
Call updateFunctionSFW("Abnormal sensation", "Abnormal", 1)
Call updateFunctionSFW("Abnormal sound", "Abnormal", 1)
Call UpdateNotSpecifiedWhere(1)
</CODE>
```

-2- Generate PDV for a particular member

This is all implemented in a class called "BoundedPDv.java." The method works as follows:

```
<CODE language="java" doctored="heavily doctored">

public void generatePdvColumns() throws DomainException {

     getPdvColumns().clear();

     generateGenderColumns(memberId);
     generateAgeColumns(memberId);
     generateSfwColumns(memberId);

    }

/** retrieve gender information from member
 *  object and update corresponding columns */
 private void generateGenderColumns(Long memberId) throws
DomainException {
     Member member = new Member(new MemberIdKey(memberId));
     String gender = member.getGender();
     if ("m".equalsIgnoreCase(gender)) {
         Long columnId = new Long(MALE_COLUMN_ID);
         setColumn(columnId, 1);
         return;
     }
     if ("f".equalsIgnoreCase(gender)) {
         Long columnId = new Long(FEMALE_COLUMN_ID);
         setColumn(columnId, 1);
         return;
     }
     Log.write(Log.ERROR,"could not determine gender of
member, got gender:" +
gender + " for memberId" + memberId + "- continuing
silently", this);

    }

 /** retrieve age information from member
```

```
  * object and update corresponding columns */
 private void generateAgeColumns(Long memberId) throws
DomainException {
     Member member = new Member(new MemberIdKey(memberId));

     int maxAge = NUM_AGE_COLUMNS*YEARS_IN_AGE_BRACKET;  //
15*7=105
     int lastAgeColumn = NUM_AGE_COLUMNS+FIRST_AGE_COLUMN-1;
 //15+3-1=17
(columnId,17) at the mo
     int age = member.getAge().intValue();


     // find the highest age bracket in which the member
     // exceeds minimum age
     int bracketMin = maxAge;
     for (int columnId=lastAgeColumn;
columnId>=FIRST_AGE_COLUMN; columnId--) {
         bracketMin=bracketMin-YEARS_IN_AGE_BRACKET;
         if (age>=bracketMin) {
          // NB.. if older than maxAge, they end up in the
highest bracket
          setColumn(new Long(columnId),1);
          return;
         }

     }

    }


/** call a stored procedure (for speed)
 * to get the columns relating to SFW information, calculate
 *  corresponding value and call setColumn to update into
pdv column list
*/
  private void generateSfwColumns(Long memberId) throws
DomainException {

         String retrieveQuery = "{call
cluster_event_severities_sp(" + memberId
+ ")}";


       while (resultSet.next()) {
          rowCount++;

          // retrieve the severity info

          int i = resultSet.getInt("i");
```

```
            int j = resultSet.getInt("j");
            int k = resultSet.getInt("k");
            int l = resultSet.getInt("l");

            // calculate value for column from these
severities
            // lose accuracy at this, the last point, in
equation
            float value = (float)calculateSfwValue(i,j,k,l);

            // and update/add this value into pdvColumnList
            Long columnId = new
Long(resultSet.getLong("columnId"));
            setColumn(columnId, value);
          }

      }


      // calculate these values once per each initialisation
of the instance
      private double aInv=1/ClusterParam.a;
      private double bInv=1/ClusterParam.b;
      private double cInv=1/ClusterParam.c;
      private double dInv=1/ClusterParam.d;

      private double calculateSfwValue(int i, int j, int k,
int l) throws
DomainException {
       return 1 + ClusterParam.upperB * (1- (Math.pow(aInv,i)
* Math.pow(bInv,j)
*Math.pow(cInv,k) *Math.pow(dInv,l)));
      }

}
```

</CODE>

for completeness the stored procedure which gets the severity i,j,k,l for the member's events is defined as follows:

<CODE language="TSQL">

```
CREATE PROCEDURE cluster_event_severities_sp
(
    @MemberId INT
)
AS
DECLARE
```

```
@num_mild       int,
@num_moderate   int,
@num_severe     int,
@num_variable   int,
@column_id      int,
@sfw_id         int
create table #temp(columnId int, SFWId int, I int, J int, K
int, L int)
declare cluster_col_cursor cursor for
    select distinct cc.ClusterColumnId, cc.SfwId
    from ClusterColumn cc,
         FormalProblem fp,
         Event e
    where e.MemberId = @MemberId
    and   e.FormalProblemId = fp.FormalProblemId
    and   fp.SFWId = cc.SFWId
    and   cc.ClusterColumnType = 'Sfw'
    and   e.OnsetSeverity in ('mild', 'moderate', 'severe',
'variable')
open cluster_col_cursor
fetch next from cluster_col_cursor into @column_id, @sfw_id
while @@FETCH_STATUS = 0
begin
    select @num_mild = count(*)
    from Event e,
         FormalProblem fp
    where e.FormalProblemId = fp.FormalProblemId
    and   e.MemberId = @MemberId
    and   fp.SfwId = @sfw_id
    and   e.OnsetSeverity = 'mild'
    select @num_moderate = count(*)
    from Event e,
         FormalProblem fp
    where e.FormalProblemId = fp.FormalProblemId
    and   e.MemberId = @MemberId
    and   fp.SfwId = @sfw_id
    and   e.OnsetSeverity = 'moderate'
    select @num_severe = count(*)
    from Event e,
         FormalProblem fp
    where e.FormalProblemId = fp.FormalProblemId
    and   e.MemberId = @MemberId
    and   fp.SfwId = @sfw_id
    and   e.OnsetSeverity = 'severe'
    select @num_variable = count(*)
    from Event e,
         FormalProblem fp
    where e.FormalProblemId = fp.FormalProblemId
    and   e.MemberId = @MemberId
    and   fp.SfwId = @sfw_id
```

```
    and   e.OnsetSeverity = 'variable'
    insert into #temp values(@column_id, @sfw_id, @num_mild,
@num_moderate,
@num_severe, @num_variable)
    fetch next from cluster_col_cursor into @column_id,
@sfw_id
end
close cluster_col_cursor
deallocate cluster_col_cursor
select * from #temp
drop table #temp

GO
```

</CODE>

-3- Calculate similarities from this PDV to other members.

This is all done inside the database.

The key code that does this is the bits of sql that follow, essentially it just implements the formula that is in the spec.

<CODE language="TSQL">

```
CREATE PROCEDURE cluster_calculate_similarities_sp
(
    @PdvIdIn int,
    @Tau     float
)
AS
DECLARE
@PdvIdOut     int
declare cluster_potential_cursor cursor for
    select distinct PdvIdOut
    from cluster_find_potential_pdv_list_view
    where PdvIdIn = @PdvIdIn
    and Tau > @Tau
delete from clusterMetric where pdvId=@pdvIdIn
open cluster_potential_cursor
fetch next from cluster_potential_cursor into @PdvIdOut
while @@FETCH_STATUS = 0
begin
    insert into ClusterMetric(PdvId, PdvId2, AmendDate, Val)
```

```
    select @PdvIdIn, @PdvIdOut, getdate(), sum(cw.weighting
* pd.Val *
pd2.Val)
    from PdvDetail pd,
         ClusterWeightings cw,
         PdvDetail pd2
    where cw.ClusterColumnId = pd.ClusterColumnId
    and   cw.ClusterColumnId2 = pd2.ClusterColumnId
    and   pd.PdvId = @PdvIdIn
    and   pd2.PdvId = @PdvIdOut
    and   cw.Weighting > @Tau
    fetch next from cluster_potential_cursor into @PdvIdOut
end
close cluster_potential_cursor
deallocate cluster_potential_cursor

GO
</CODE>
```

The above code depends on "cluster_find_potential_pdv_list_view" which is a view used, for speed purposes only, to create virtual subset of all pdvs. (Ie only the pdv-pdv matches where the similarity is >0 get a value inserted)

That view is defined as follows:

```
<CODE language="TSQL">

CREATE PROCEDURE cluster_find_potential_pdv_list_sp
(
    @PdvId int,
    @Tau   float
)
AS
insert into #potential_pdv
select distinct p2.PdvId
from Pdv p,
     PdvDetail pd,
     ClusterWeightings cw,
     PdvDetail pd2,
     Pdv p2
where pd.PdvId = p.PdvId
and   cw.ClusterColumnId = pd.ClusterColumnId
and   cw.ClusterColumnId2 = pd2.ClusterColumnId
and   pd2.PdvId = p2.PdvId
and   p.PdvId = @PdvId
and   p2.isDefault='Y'
and   cw.Weighting > @Tau
```

```
GO
```

</CODE>

4- Find the top N members:

**This is pretty simple really...
Essentially we just iterate through the list of pdvs starting at the most similar until we get to the nth member. At that point we have a value which can be used to select out the speific members via code which says basically "get all members where the similarity value > @calculatedMinValue " to get our N members.**

<CODE language="TSQL">

```
CREATE PROCEDURE cluster_find_value_sp
(
@pdvID INT,
@n INT
)
AS
declare
@tmpVal as float,
@curVal as float,
@cnt as int
set @curval = 0
set @cnt = 0
--create cursor
DECLARE  val_cursor CURSOR FOR
     SELECT    val from clustermetric
     WHERE     pdvid = @pdvID
     ORDER BY val desc
--search for nth value
--search for null values??
OPEN  val_cursor
FETCH NEXT FROM  val_cursor INTO @curVal
SET @cnt = @cnt + 1
while @@FETCH_STATUS = 0 AND @cnt < @n-1
begin
     FETCH NEXT FROM  val_cursor INTO @curVal
     SET @cnt = @cnt + 1
end
print @curVal
CLOSE  val_cursor
DEALLOCATE  val_cursor
return @curVal
```

```
GO
</CODE>
```

The foregoing description of the preferred embodiments of this invention has been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed, and obviously, many modifications and variations are possible, such as different listings (and thus divisions of the semantic plane) of the SFW's, available reportable problems and formal problems, different subject matter than human medical systemic states of being being encoded and mined, etc.. Such modifications and variations that may be apparent to persons skilled in the art are intended to be included within the scope of this invention as defined by the accompanying claims.